We Claim:

1. A system comprising:

a central processing unit; and

a hardware accelerator operably connected to the central processing unit, the hardware accelerator adapted to translate stack-based instructions into register-based instructions native to the central processing unit.

2. The system of Claim 1, wherein the stack-based instructions are associated with a virtual machine.

3. The system of Claim 1, wherein the stack-based instructions are Java bytecode.

4. The system of Claim 1, wherein the hardware accelerator implements at least part of a Java virtual machine.

5. The system of Claim 1, wherein the hardware accelerator is connected between a memory and the central processing unit.

6. The system of Claim 5, wherein the hardware accelerator is connected between an instruction cache and the central processing unit.

7. The system of Claim 1, wherein the hardware accelerator is adapted to manage a java stack.

8. The system of Claim 1, wherein the hardware accelerator is adapted to store at least some of a Java operand stack in a register file connected to the central processing unit.

9. The system of Claim 8, wherein the hardware accelerator has access to the data bus of the central processing unit.

10. The system of Claim 8, wherein the hardware accelerator is adapted to swap parts of the operand stack are in and out of the register file from a memory.

11. The system of Claim 8, wherein the central processing unit is operably connected to a native register file and a register file controlled by the hardware accelerator

12. The system of Claim 11, wherein the at least some of the Java operand stack is stored in the register file controlled by the hardware accelerator.

13. The system of Claim 8, wherein the central processing unit is operably connected to a native register file and wherein the at least some of the Java operand stack is stored in the native register file.

14. The system of Claim 8, wherein the hardware controller is further adapted to store at least some variables in the register file.

15. The system of Claim 8, wherein the hardware accelerator is incorporated within the central processing unit.

16. The system of Claim 1, wherein the hardware accelerator has access to at least one bus of the central processing unit.

17. The system of Claim 1, wherein the hardware accelerator is adapted to examine the stack-based instructions to determine whether multiple stack-based instructions can be combined into fewer register-based instructions.

18. The system of Claim 17, wherein multiple stack-based instructions pass through the hardware accelerator concurrently to allow for the operation of the combining logic.

19. The system of Claim 1, wherein the hardware accelerator is divided into pipelined stages.

20. The system of Claim 1, wherein the hardware accelerator is adapted to be flushed under predetermined conditions.

21. The system of Claim 1, wherein the central processing unit and hardware accelerator are on the same chip.

22. The system of Claim 1, wherein the hardware accelerator produces an exception upon at least one of the stack-based instructions, and wherein the central processing unit will, in software, translate the at least one of the stack-based instructions causing the exception.

23. The system of Claim 1, wherein the hardware accelerator is incorporated within the central processing unit.

24. A system comprising:

a central processing unit; and

a hardware java accelerator operably connected to the central processing unit, the hardware java accelerator adapted to translate java bytecodes into instructions native to the central processing unit.

25. A system comprising:

a central processing unit; and

a hardware accelerator operably connected to the central processing unit, the hardware accelerator adapted implement at least part of a virtual machine associated with a computer language, the hardware accelerator adapted to translate instructions for the virtual machine into native instructions for the central processing unit.

26 . A method comprising:

moving a stack-based instruction from memory to a hardware accelerator;

in the hardware accelerator, converting the stack-based instruction into a register-based instruction native to a central processing unit; and

in the central processing unit, executing the register-based instruction.

27. The method of Claim 26, wherein the stack-based instructions are associated with a virtual machine.

28. The method of Claim 26, wherein the stack-based instructions are Java bytecode.

29. The method of Claim 26, wherein the accelerator implements at least part of a Java virtual machine.

30. The method of Claim 26, further comprising, in the hardware accelerator, managing a java stack.

31. The method of Claim 26, further comprising storing at least some of a Java operand stack in a register file connected to the central processing unit.

32. The method of Claim 26, wherein parts of the operand stack are swapped in and out of the register file from a memory by the hardware accelerator.

33. The method of Claim 26, wherein the hardware accelerator examines the stack-based instructions to determine whether multiple stack-based instructions can be combined into fewer register-based instructions.

34. The method of Claim 26, further comprising producing an exception in the hardware accelerator upon at least one stack-based instruction, and translating the at least one stack-based instruction causing the exception in software in the central processing unit.

35. The method of Claim 26, wherein the central processing unit and hardware accelerator are on the same chip.

36. The method of Claim 26, wherein the hardware accelerator is incorporated within the central processing unit.